Microsoft®
Customer Care Framework 2009

Product brief

Microsoft® Corporation
Published: October 2008

Abstract

Built on the foundation of Customer Care Framework (CCF) 2008, CCF 2009 introduces several enhancements to quickly compose UI components and services spanning multiple interaction channels.

CCF particularly fits to the following scenarios:

- **Delivery of composite applications** such as the teller desktop in banking or Customer Service Representatives (CSR) desktop in contact centers. This scenario is also appropriate to extend legacy applications by adding new features or to adapt existing processes into a new technology.

- **Multi-channel infrastructures to support:**
    - Cross-Channel business processes
    - Unified access to customer information spread across different systems

- **Development of new multi-channel front-end** obeying an IT strategy heading towards a Service Oriented Architecture (SOA).

The information contained in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation. Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property. The example companies, organizations, products, people and events depicted herein are fictitious. No association with any real company, organization, product, person or event is intended or should be inferred. © 2008 Microsoft Corporation. All rights reserved. Microsoft Customer Care Framework is a trademark of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Table of Contents

1 Microsoft Customer Care 2009 Overview 4
2 CCF 2009 architecture 4
2.1 CCF Server components 5
2.1.1 CCF Core Services 5
2.1.2 CCF Distributed Connectivity Services (DCS) 6
2.1.3 Configuration Server 7
2.1.4 CCF Databases 7
2.2 CCF Client Components 7
2.2.1 Application Integration Framework (AIF) 7
2.2.2 Hosted Application Toolkit (HAT) 8
2.2.3 Multi-channel Engine (MCE) 9
2.2.4 DCS Service Agent 9
2.3 CCF Management Solutions 9
2.4 CCF Developer tools 10
3 CCF 2008 and 2009 feature comparison 10
4 Product requirements 12
5 Product dependencies 13
6 Further information 13

Copyright © 2008 Microsoft Corporation

1 Microsoft Customer Care 2009 Overview

The Microsoft Customer Care 2009 (CCF) solution is an end-to-end application infrastructure for delivery of composite applications. It includes development and runtime components. Applications built with CCF can provide unified access to customer information spread across different systems and aggregate different modes of customer interactions (channels).

Unlike expensive, rip and replace, and risky investments on custom development, CCF provides non-intrusive integration with existing systems.

The core characteristics of the CCF solution are:

- **User Interface composition** – CCF allows the creation of composite UI applications, with support for enterprise level patterns such as Model-View-Controller (MVC).
- **Distributed architecture** – CCF allows the creation of centrally managed distributed components. It helps in reducing latency, infrastructural costs and enabling offline operations.
- **Developer agility** – CCF employs modern process driven development methodologies empowering developers through toolkits to reduce application development time while increasing quality and standardization.
- **Integration and automation** – CCF provides non-intrusive application integration and automation both on the front-end (UI automation) and back-end (Service composition).
- **Security** – CCF employs modern standards for securing communications such as WS-Security, WS-Trust, WS-SecureConversation, Security Assertion Markup Language (SAML) and Kerberos. CCF authorization model allows centrally managed role-based administration. CCF also provides E-SSO services to allow mapping of user identities across systems.
- **Scalability and reliability** – CCF is designed to scale vertically and horizontally and can be used in clustered and load balanced scenarios.

2 CCF 2009 architecture

CCF 2009 consists of four major subsystems:

1) CCF Middleware Server Components
2) CCF Front-end Tools and Client Components
3) CCF Management Solutions
4) CCF Developer Tools

Copyright © 2008 Microsoft Corporation

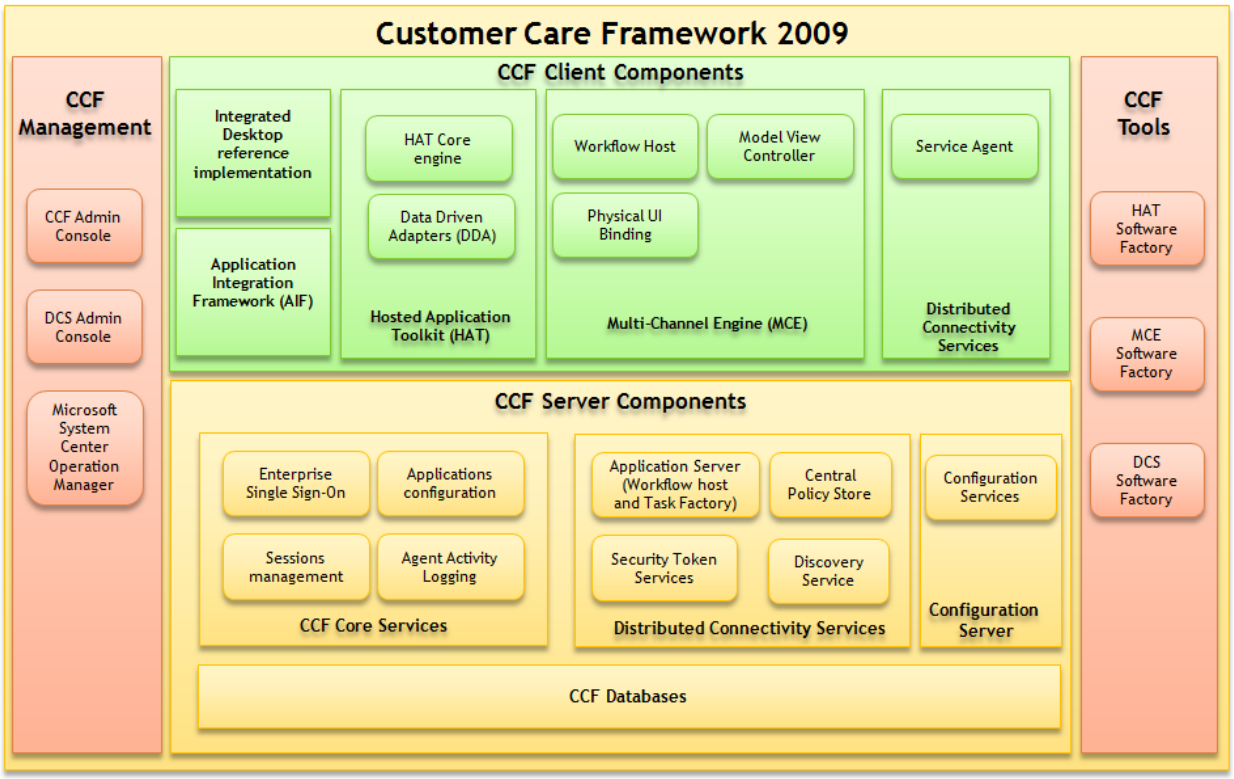

Figure 1: CCF 2009 Architecture

2.1 CCF Server components

2.1.1 CCF Core Services

The **CCF Core Services** performs logging and reporting, configuration, session management, SSO credential storing, and other functions. It supports horizontal scale-out, clustered and load-balanced deployment.

Some of the notable highlights of the CCF Server include:

- **Agent Activity Logging** provides records of transactions which can be aggregated in customizable reports to enable extensive business analysis e.g. to optimize Contact Center processes and customer handling.

- CCF uses **Enterprise Single Sign-On (E-SSO)** and the Windows Active Directory® service to integrate authentication and authorization for agent roles and applications. E-SSO provides a way to map a Windows user ID to non-Windows user credentials. This is a key feature for businesses that aim to aggregate multiple applications spread over diverse systems.

Copyright © 2008 Microsoft Corporation

- **Sessions management** with session transfer supporting automatic application launching. If a step in a workflow process needs to be continued by another agent or at a later time, the session context, including the state of the supporting applications and information can be automatically transferred to the agent, restored later or transferred to another channel.

- **Applications configuration**. All configuration information is stored on a core server. In the event that an agent's workstation fails, the agent's applications and operating state can be restored when the agent logs back on from another workstation.

2.1.2 CCF Distributed Connectivity Services (DCS)

Distributed Connectivity Services (DCS) is an application infrastructure providing an end-to-end development model and reference architecture that enables a business-driven development model.

DCS provides service management capabilities such as transparent service location, dynamic client, context driven business process selection, scalable hosting of workflows and service configuration and a standards based security token service. DCS uses Microsoft .NET 3.0™, Windows® Communication Foundation (WCF) and Windows Workflow Foundation (WF) as its foundation. DCS includes reusable Application Blocks in a Software Factory to simplify solution development and deployment of processes and services.

DCS consists of the following Application Blocks:

- **Discovery Service Application Block** – Allows client applications to discover at run time the service endpoints based on service type. DCS utilizes a Microsoft SQL Server® based repository store and a WCF based service discovery method (WS-Discovery).
- **Central Policy Store Application Block** – It provides centralized configuration of service policies. Policies are consumed by the client application through metadata exchange.
- **DCS application server** - The DCS application server is hosted inside Microsoft Internet Information Server and is composed of two main components:
    - **Task Factory** – Supports DCS Business Logic selection based on a context.
    - **Workflow Host Application Block** – Executes Windows Workflow-based Business Logic.
- **Security Token Service Application Block** – Security Token Service implements the WS-Trust based authority service that issues SAML based tokens. Allows the configuration of external claim providers caching of SAML tokens for increased performance. Windows-based authentication of clients leverages Microsoft Active Directory.

DCS also provides the following patterns:

- **DCS Dynamic Messaging Patterns** – DCS provides unique messaging patterns for distributed services in an enterprise to communicate with each other. These patterns include messaging to facilitate the client applications to discover the service location and communication policies at runtime, transportation of context data, authentication, and authorization. DCS Messaging

Copyright © 2008 Microsoft Corporation

Patterns leverage Windows Communication Foundation and have been built on WS-* standards in order to maximize interoperability and extensibility.

- **Dynamic Service Policy Patterns** – DCS facilitates centrally storing and managing the service policies. When the DCS based service starts, it contacts the central policy service, retrieves the relevant policies and configures its WSDL and receive pipeline according to the policies.

2.1.3 Configuration Server

The CCF Configuration Server allows developers to dynamically create .NET configuration files from hierarchical sets of assertion about an application context. These assertions may be related to user roles, business environments or geographical locations for example.

CCF also provides the **ConfigurationUpdater** client API that allows interaction with the configuration server

2.1.4 CCF Databases

The CCF Data layer is used to store configuration and audit information (activities performed by a CCF Client component). CCF requires the Databases to be hosted in Microsoft SQL Server to host the Databases.

CCF also allows generating reports from the persisted audit information with SQL Reporting Services.

2.2 CCF Client Components

2.2.1 Application Integration Framework (AIF)

The Application Integration Framework (AIF) is a software application framework that facilitates information sharing, interaction and communication between applications and processes running on the client. It is built upon the "Pattern & Practices" Composite UI Application Block (CAB). AIF leverages the CAB utilities and services for messaging, visual containment, application loading, state management, etc.

In CCF, AIF is used by the integrated desktop: a Windows Smart Client that runs on the Customer Service Representatives (CSRs) workstation within Microsoft's Windows XP® or Windows Vista® operating system.

The **Customer Care Framework Integrated Desktop (ID)** is a reference implementation that enables the reuse of existing line-of-business applications. It provides a composite user interface that clearly shows how applications and services can work together within shared workflows, computer telephony integration (CTI), and session management. Some of the key features:

- ID optimizes the CSR's desktop by combining existing resources into a single user interface. The unified user interface that ID provides enables CSRs to work more efficiently and with higher accuracy.
- ID is highly customizable. The source code supplied with CCF is a reference implementation that enables the development of user interfaces that meet specific business requirements.

Copyright © 2008 Microsoft Corporation

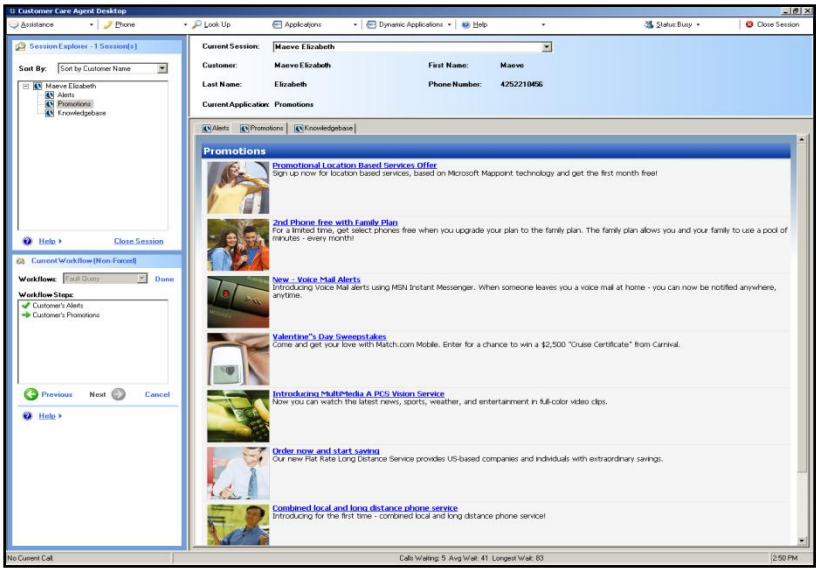
Customer Care Agent Desktop
Promotions
Promotional Location Based Services Offer
2nd Phone free with Family Plan
New - Voice Mail Alerts
Valentine's Day Sweepstakes
Introducing MultiMedia A PCS Vision Service
Order now and start saving
Combined local and long distance phone service

Figure 2: CCF 2009 Integrated Desktop reference implementation

2.2.2 Hosted Application Toolkit (HAT)

The Hosted Application Toolkit (HAT) significantly reduces the effort required to integrate applications in CCF. It allows applications to be hosted and automated in the Integrated Desktop with minimal coding. It provides an Application inspector tool, a set of Data Driven Adapters (DDA) and a visual designer that is used to create the business logic called *Automation (click button A, copy data from textbox and paste it to…)*, between hosted applications . This way the process of integrating an application is the execution of three well defined steps: Inspect, Automate, Host.

HAT has three major components:

1) **Application Software Factory**, a tool which allows visual inspection of the UI controls and generates an XML description file of the controls that will participate in the automation step, performed in a Visual Studio environment.
2) **HAT Core Engine,** drives automation leveraging Windows Workflow Foundation.
3) **Data Driven Adapters (DDA)** provides generic access to user interfaces through XML documents. CCF2009 provides DDAs for Windows-based applications, Web applications and Java-based applications.

Copyright © 2008 Microsoft Corporation

2.2.3 Multi-channel Engine (MCE)

The **Multi-channel Engine (MCE)** in CCF 2009 provides a common engine that allows UI processes to share common logic and metadata on different delivery channels. The MCE provides two major functionalities:

1) Separation between channel independent and channel dependent development. The channel independent components include all the logical aspect of User Interface, the navigational control logic, the logical information/data views and event handling. The channel dependent UI provides the actual navigational controls and visual representation of the information.

   The front-end multi-channel engine improves the consistency of the navigational logic amongst different delivery channels, increasing the productivity of the cross channels UI design. The standardization and reuse of the navigational logic will allow the developer of a particular channel UI to mainly focus on the particular physical layout and specific navigation of the channel.

2) The ability to persist an UI process state and to move it between channels. This represent a significant improvement in the user experience because it allows the interrupted acquisition of data without have the need for the user to start the process again all the times. It also enables the resuming of sessions across different delivery channel seamlessly!

2.2.4 DCS Service Agent

The **DCS Service Agent Application Block** is a client component that uses the DCS discovery service to identify the target services endpoints and performs metadata exchange to determine service policies. It dynamically builds the send-pipeline as requested by service policies, including the standard DCS ones: encoding, security, client caching, logging and context transportation.

2.3 CCF Management Solutions

The CCF management solution consists of three components – CCF Admin Console, DCS Admin Console and CCF Management Pack for System Centre.

The **CCF Admin Console** allows an administrator to centrally configure all aspects of the Integrated Desktop running in an enterprise environment. The Console allows management of client configuration and runtime update to all clients and also allows federation of different components of the integrated desktop.

The **DCS Admin Console** allows administrators to control all aspects of service based communication and DCS business logic. It enables to deploy services and tasks, add and edit Task Factory filters, change Discovery configured instances and policy on each instance.

The **CCF Management Pack for Microsoft System Center Operations Manager** provides monitoring of business processes and infrastructure services through Microsoft Operations Manager 2007™. This pack also allows centralized management of CCF infrastructure components through Microsoft's System Center suite.

Copyright © 2008 Microsoft Corporation

2.4 CCF Developer tools

CCF provides a set of tools to facilitate rapid development of customer care solutions.

1. **DCS Software Factory** is a set of Microsoft Visual Studio add-ins and Guidance automation extensions that enable rapid development of workflow-enabled services using proven best practices. The Factory supports the creation of WCF based Messages, conversational workflows, service stubs from workflows and DCS service agents. It also helps in defining service contracts and services as workflows.

2. **MCE Software Factory** is a set of Visual Studio add-ins and Guidance automation extensions that enable the creation of logical views and UI process workflows, the creation of a physical representation of logical views for Winforms and asp.net.

3. **HAT Software Factory** is a set of Visual Studio add-ins and Guidance automation extensions that enable the inspection of existing user interfaces such as windows and web applications and the creation of UI automations as workflows. It also enables automatic deployment of the toolkit artifacts.

3 CCF 2008 and 2009 feature comparison

Components	Features	Description	CCF 2008	CCF 2009
CCF core	Authorization provider	Allows role based authorization for CCF clients	X	X
	Logging Provider	Allows CCF logging through different providers (file, event log…)	X	improved
	Security extensions	Allows CCF Clients to operate in different domains	X	X
	SSO Provider	Allows mapping of user identities across systems	X	X
	Audit provider	Allows recording of an agent's interaction with customers and hosted applications	X	improved
	CTI provider	Allows connectivity to CTI systems	X	X
	Application Integration Framework (AIF)	Client event bus for messaging, visual containment, application loading, state management…	X	improved
	CITRIX Generic stub	Allows hosting of Citrix applications	X	improved
HAT	Win32 DDA	Provides generic access automations to win32	X	X

Copyright © 2008 Microsoft Corporation

		applications		
	Web DDA	Provides generic access automations to web applications	X	improved
	Java DDA	Provides generic access automations to Java applications		X
	Automation Manager & Designer	Design and drives the automations between hosted applications through DDAs.	X	X
	Application Inspector	Allows visual inspection of existing user interfaces		X
Distributed Connectivity Services (DCS)	DCS Dynamic Messaging Patterns	Messaging patterns for distributed services		X
	Dynamic Service Policy Patterns	Stores and manages the service policies		X
	Discovery Service Application Block	Allows client applications to discover at run time the service endpoints based on service type		X
	Central Policy Store Application Block	Centralized configuration of service policies		X
	Task factory	Supports business Logic selection based on the context.		X
	Workflow Host Application Block	Executes Windows Workflow based Business Logic		X
	Security Token Service Application Block	Allows configuring of external claim providers.		X
	Service Agent application block	Consumes discovery service to identify the target service endpoint and performs metadata exchange to determine service policies		X
Multi-channel Engine (MCE)	Workflow Host Application Block	Executes MCE based Business Process		X
	Model View Controller application block	Allows separation of logical and physical views		X
	Physical UI binding system	Allows the data binding to physical views.		X
Tools	MCE Software Factory	Visual Studio add-ins and Guidance automation		X

Copyright © 2008 Microsoft Corporation

		for MCE development		
	HAT Software Factory	Visual Studio add-ins and Guidance automation for HAT development	X	Improved
	DCS Software Factory	Visual Studio add-ins and Guidance automation for DSC development		X
Configuration Server	Configuration Services	Dynamic composition of configuration files		X
Reference implementation		Sample solutions to demonstrate how the various features work	X	improved

4 Product requirements

	Operating Systems	**32/64 bits**	**Product dependencies**
CCF Core Server	Windows Server 2003 Service Pack 2 and above Windows Server 2008	32bits /64 bits	.NET Framework 3.0 or 3.5 IIS 6.0 or higher
CCF database server	Windows Server 2003 Service Pack 2 and above Windows Server 2008	32 bits/64 bits	SQL Server 2005 Service Pack 2 standard edition or higher
CCF client workstation	Windows XP Service Pack 2 and above Windows Vista SP1	32 bits	.NET Framework 3.0 or 3.5 SQL Server Compact Edition 3.5
CCF developer workstation	Windows Server 2003 with the latest SP Windows Server 2008. Windows Vista	32 bits/64 bits	.NET Framework 3.0 or 3.5 Visual Studio 2005 SP1 or Visual Studio 2008 Visual Studio 2005 SDK Version 4.0 or Visual Studio 2008 SDK Version 1.0 Guidance Automation

Copyright © 2008 Microsoft Corporation

			Extensions 1.4 (GAX) DSL tools. DSL Tools are not required for if you have Visual Studio 2008 installed. Windows Workflow Foundation Extensions for Visual Studio 2005
CCF Application Server	Windows Server 2003 Service Pack 2 and above Windows Server 2008	32bits /64 bits	.NET Framework 3.0 or 3.5 IIS 6.0 or more

5 Product dependencies

The Java Accessibility Bridge (JAB) is required by the CCF developer environment and the CCF client environment if the Java DDAs are going to be used to host Java applications. The JAB is available from Sun Microsystems.
http://java.sun.com/javase/technologies/accessibility/accessbridge/index.jsp

The Java Runtime Environment (JRE) is required by the CCF Developer environment and the CCF client environment to run Java applications. The current version supported by CCF is JRE 1.6.
https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=jre-6u6-oth-JPR@CDS-CDS_Developer

CITRIX ICA client version 7 and above (Virtual Channel support is mandatory) and CITRIX presentation Server 4.0 or 4.5 are required to host Citrix applications inside CCF Integrated desktop reference implementation.

6 Further information

For more information about Microsoft Customer Care Framework refer to
http://www.microsoft.com/ccf

For more information about development with Microsoft Customer Care Framework refer to
http://msdn.microsoft.com/en-us/isg/bb421305.aspx

For more information about security specifications (WS-Security…) refer to
http://msdn.microsoft.com/en-us/library/ms951273.aspx

Copyright © 2008 Microsoft Corporation

For more information about Composite Application block (CAB), refer to http://msdn.microsoft.com/en-us/library/aa480450.aspx

For more information about Windows Communication Foundation (WCF) refer to http://msdn.microsoft.com/en-us/netframework/aa663324.aspx

For more information about Windows Workflow Foundation (WF), refer to http://msdn.microsoft.com/en-us/netframework/aa663328.aspx

For more information about Microsoft Enterprise Single Sign On (E-SSO), refer to http://msdn.microsoft.com/en-us/library/aa745042.aspx

For more information about ClickOnce Deployment, refer to http://msdn.microsoft.com/en-us/library/aa745042.aspx

For Microsoft SQL Reporting Server system requirements, refer to http://www.microsoft.com/sql/technologies/reporting/default.mspx

For more information about Visual Studio Extensions, refer to http://www.microsoft.com/downloads/details.aspx?FamilyId=5D61409E-1FA3-48CF-8023-E8F38E709BA6&displaylang=en

For more information about Domain-Specific Language Tools for Visual Studio 2005, refer to http://www.microsoft.com/downloads/details.aspx?familyid=693EE22D-4BB1-450D-835C-59EEBCB9F2AE&displaylang=en

Copyright © 2008 Microsoft Corporation